fraccap, *fc*		fractional capability variable		
expression, x, a, b		expression variable		
terminals	::=			
	|	λ		
	|	$\otimes$		
	|	$\multimap$		
	|	$\vdash$		
	|	$\in$		
	|	$\forall$		
	|	Cap		
	|	Type		
f	::=			fractional capability
	|	fc		variable
	|	**Zero**		zero
	|	**Succ** f		successor
t	::=			linear type
	|	1		unit
	|	$t \otimes t'$		pair
	|	$t \multimap t'$		arrow
	|	$\forall fc.t$	bind fc in t	fractional capability abstraction
	|	**Arr** $[f]$		array
	|	$t\{f/fc\}$	M	substitution
p	::=			primitive
	|	**share**		share array
	|	**combine**		unshare array
	|	**free**		free arrary
	|	**copy**		copy array
	|	**swap**		swap array
	|	**asum**		$\sum_i \lvert x_i \rvert$
	|	**axpy**		$x := \alpha x + y$
	|	**dot**		$x \cdot y$
	|	**nrm2**		$\lVert x \rVert^2$
	|	**rot**		plane rotation
	|	**rotg**		Givens rotation
	|	**rotm**		modified givens rotation
	|	**rotmg**		generate modified Givens rotation
	|	**scal**		$x := \alpha x$
	|	**iamax**		index of maximum absolute value
	|	**iamin**		index of minimum absolute value (Intel only)
e	::=			expression
	|	x		variable
	|	$()$		unit introduction
	|	**let** $() = e$ **in** e'		unit elimination
	|	(e, e')		pair introduction
	|	**let** $(a, b) = e$ **in** e'	bind $a \cup b$ in e'	pair elimination

	$\lambda x : t.e$	bind x in e	abstraction
	$e\ e'$		application
	$\mathbf{Array}\ (e)$		array introduction
	$\mathbf{let}\ x = e\ \mathbf{in}\ e'$	bind x in e'	array elimination
	p		Level 1 BLAS routine primitives
	$\forall\, fc.e$		frac cap abstraction
	$e[f]$		frac cap specialisation

Θ	::=	fractional capability environment
	$\cdot$	
	Θ, fc	

$\Gamma,\ \Delta$	::=	linear types environment
	$\cdot$	
	$\Gamma, x : t$	
	Γ, Δ	

formula	::=	
	judgement	
	$x : t\ \in\ \Gamma$	
	$fc\ \in\ \Theta$	

Well_Formed	::=	
	$\Theta \vdash f\ \mathsf{Cap}$	Valid fractional capabilities
	$\Theta \vdash t\ \mathsf{Type}$	Valid types

Types	::=	
	$\Theta; \Gamma \vdash e : t$	Tying rules for expressions

judgement	::=	
	Well_Formed	
	Types	

user_syntax	::=	
	fraccap	
	expression	
	terminals	
	f	
	t	
	p	
	e	
	Θ	
	Γ	
	formula	

$\boxed{\Theta \vdash f\ \mathsf{Cap}}$ Valid fractional capabilities

$$\frac{fc\ \in\ \Theta}{\Theta \vdash fc\ \mathsf{Cap}}\ \text{WF\_Cap\_Var}$$

$$\frac{}{\Theta \vdash \mathbf{Zero}\ \mathsf{Cap}}\ \text{WF\_Cap\_Zero}$$

$$\frac{\Theta \vdash f\ \mathsf{Cap}}{\Theta \vdash \mathbf{Succ}\, f\ \mathsf{Cap}}\ \textsc{WF_Cap_Succ}$$

$\boxed{\Theta \vdash t\ \mathsf{Type}}$ Valid types

$$\frac{}{\Theta \vdash 1\ \mathsf{Type}}\ \textsc{WF_Type_Unit}$$

$$\frac{\Theta \vdash t\ \mathsf{Type} \quad \Theta \vdash t'\ \mathsf{Type}}{\Theta \vdash t \otimes t'\ \mathsf{Type}}\ \textsc{WF_Type_Pair}$$

$$\frac{\Theta \vdash t\ \mathsf{Type} \quad \Theta \vdash t'\ \mathsf{Type}}{\Theta \vdash t \multimap t'\ \mathsf{Type}}\ \textsc{WF_Type_Lollipop}$$

$$\frac{\Theta \vdash f\ \mathsf{Cap}}{\Theta \vdash \mathbf{Arr}\,[f]\ \mathsf{Type}}\ \textsc{WF_Type_Array}$$

$$\frac{\Theta, fc \vdash t\ \mathsf{Type}}{\Theta \vdash \forall\, fc.t\ \mathsf{Type}}\ \textsc{WF_Type_ForAll}$$

$\boxed{\Theta; \Gamma \vdash e : t}$ Tying rules for expressions

$$\frac{}{\Theta; \cdot, x : t \vdash x : t}\ \textsc{Ty_Var}$$

$$\frac{}{\Theta; \cdot \vdash () : 1}\ \textsc{Ty_Unit_Intro}$$

$$\frac{\Theta; \Gamma \vdash e : 1 \quad \Theta; \Delta, x : 1 \vdash e' : t}{\Theta; \Gamma, \Delta \vdash \mathbf{let}\, x = e\, \mathbf{in}\, e' : t}\ \textsc{Ty_Unit_Elim}$$

$$\frac{\Theta; \Gamma \vdash e : t \quad \Theta; \Delta \vdash e' : t'}{\Theta; \Gamma, \Delta \vdash (e, e') : t \otimes t'}\ \textsc{Ty_Pair_Intro}$$

$$\frac{\Theta; \Gamma \vdash e_{12} : t_1 \otimes t_2 \quad \Theta; \Delta, a : t_1, b : t_2 \vdash e : t}{\Theta; \Gamma, \Delta \vdash \mathbf{let}\, (a, b) = e_{12}\, \mathbf{in}\, e : t}\ \textsc{Ty_Pair_Elim}$$

$$\frac{\Theta; \Gamma, x : t' \vdash e : t \quad \Theta \vdash t\ \mathsf{Type}}{\Theta; \Gamma \vdash \lambda x : t'.e : t' \multimap t}\ \textsc{Ty_Lambda}$$

$$\frac{\Theta; \Gamma \vdash e : t' \multimap t \quad \Theta; \Delta \vdash e' : t'}{\Theta; \Gamma, \Delta \vdash e\ e' : t}\ \textsc{Ty_App}$$

$$\frac{\Theta; \Gamma \vdash e : \mathbf{Arr}\,[f]}{\Theta; \Gamma \vdash \mathbf{Array}\,(e) : \mathbf{Arr}\,[\mathbf{Zero}]}\ \textsc{Ty_Array_Intro}$$

$$\frac{\Theta; \Gamma \vdash e : \mathbf{Arr}\,[f] \quad \Theta; \Delta, x : \mathbf{Arr}\,[f] \vdash e' : t'}{\Theta; \Gamma, \Delta \vdash \mathbf{let}\, x = e\, \mathbf{in}\, e' : t'}\ \textsc{Ty_Array_Elim}$$

$$\frac{\Theta, fc; \Gamma \vdash e : t \quad \Theta, fc \vdash t\ \mathsf{Type}}{\Theta; \Gamma \vdash \forall\, fc.e : \forall\, fc.t}\ \textsc{Ty_ForAll_frac_cap}$$

$$\frac{\Theta; \Gamma \vdash e : \forall\, fc.t}{\Theta; \Gamma \vdash e[f] : t\{f/fc\}}\ \textsc{Ty_Specialise_frac_cap}$$

```
Definition rules:         19 good    0 bad
Definition rule clauses: 43 good    0 bad
```